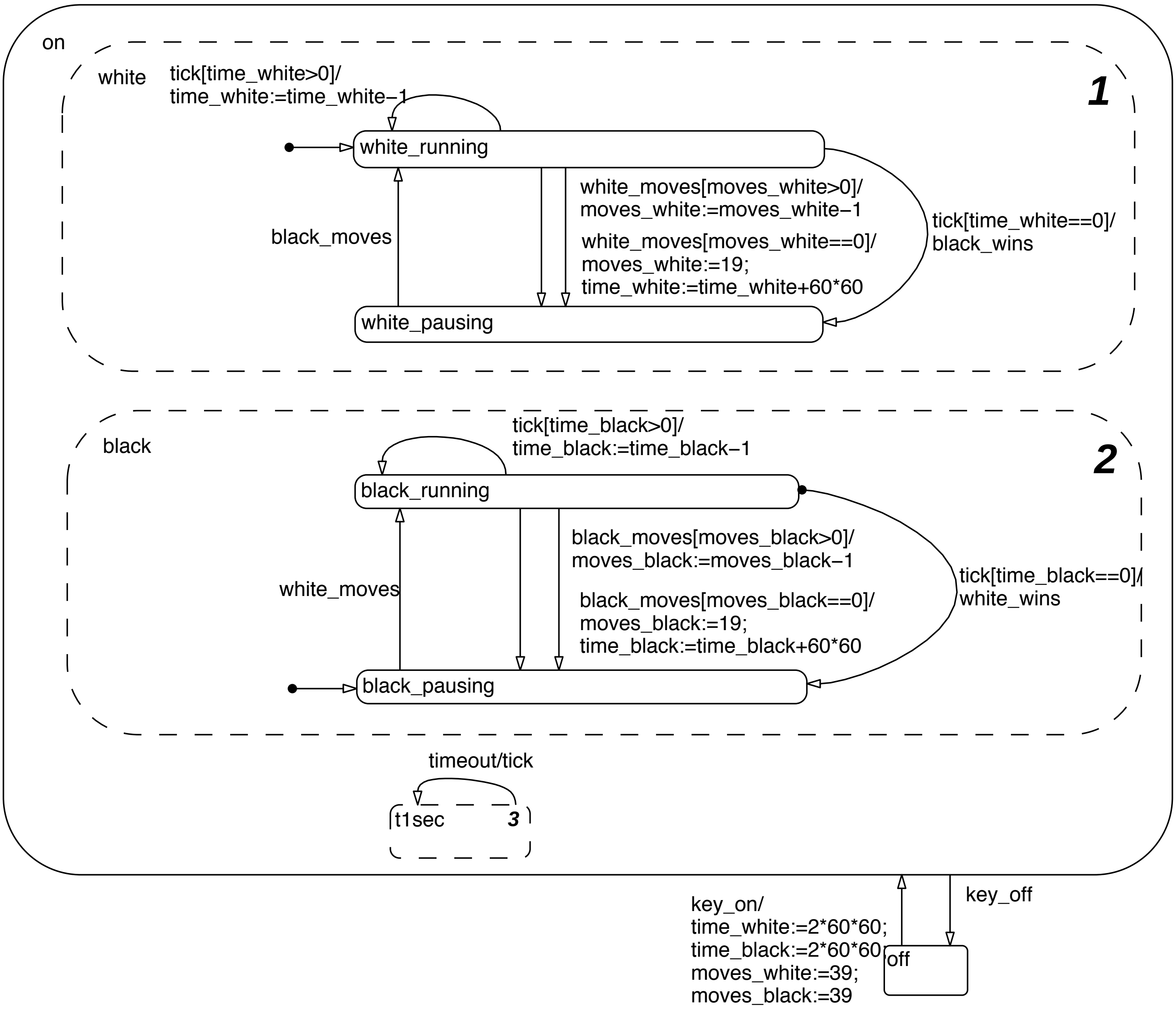
Prof. Bernd Finkbeiner
Dr. Calogero Zarba
Moritz Hahn

Sample Solution for Problem Set 2
Due: May 2, 2007, 11:14 am

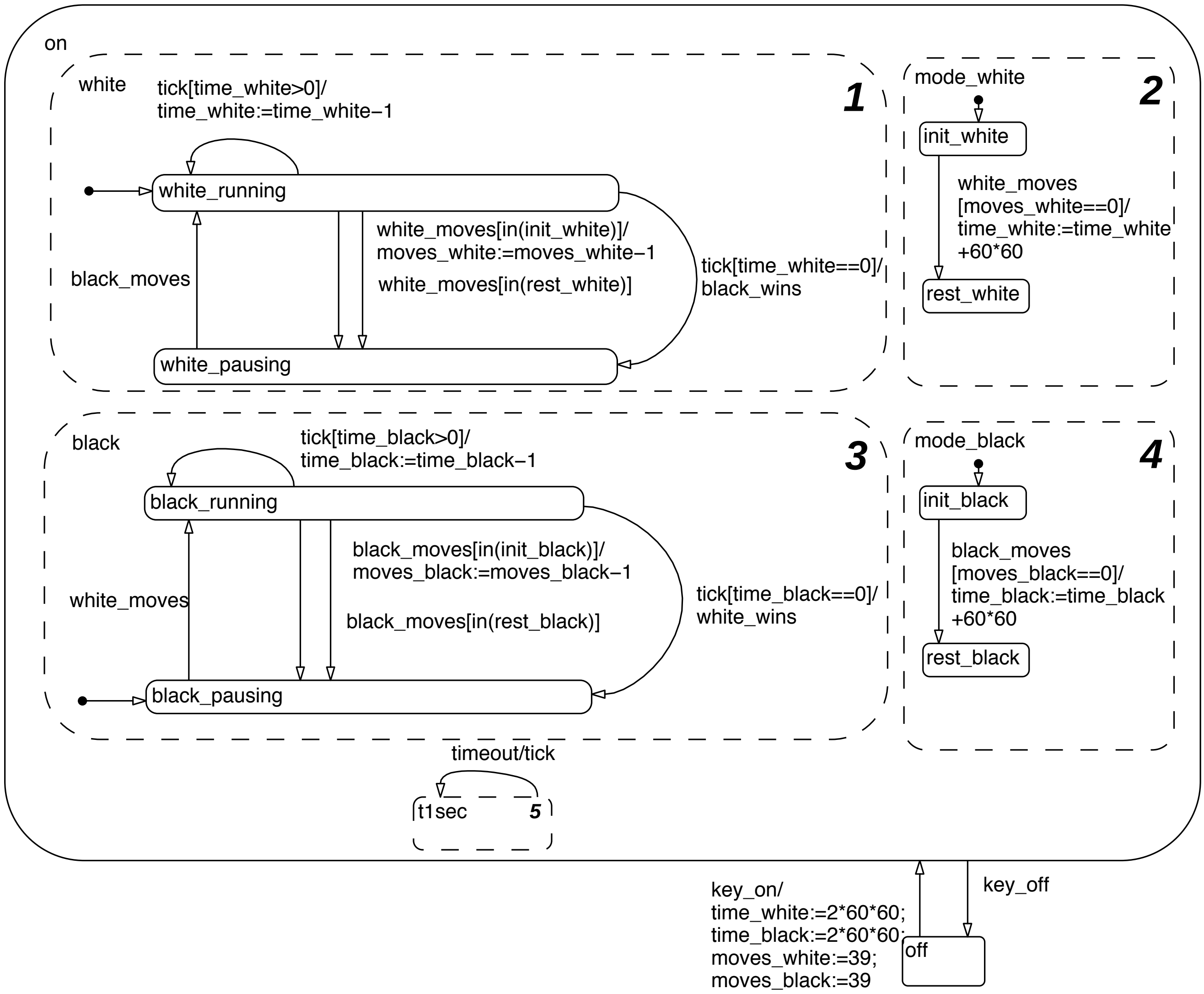
Embedded Systems

Problem 1 (Statecharts) 60 points

For problems (a), (b) and (c) we actually meant to increment the time of the clocks and not to set them.
(a)

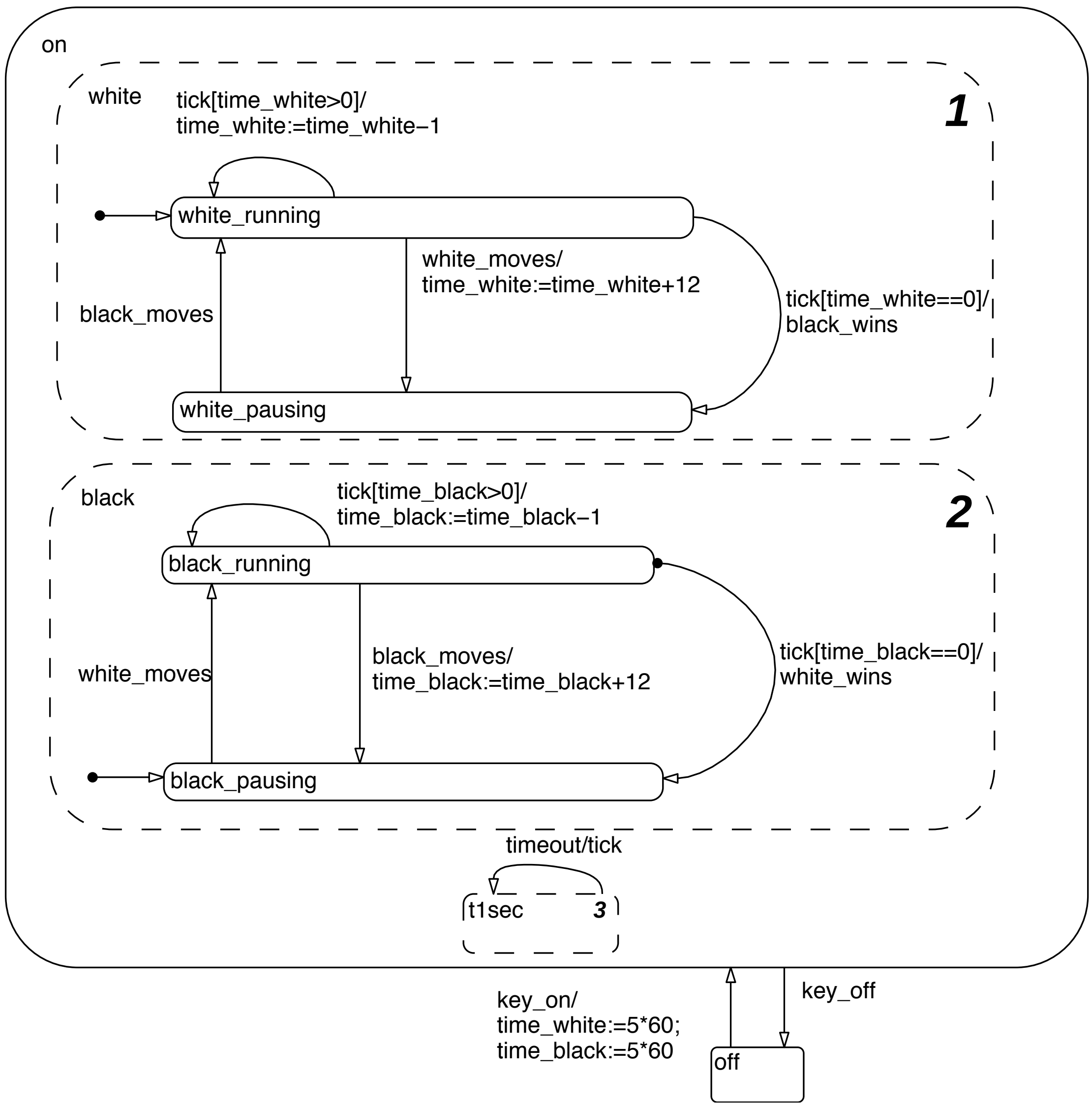


(b)

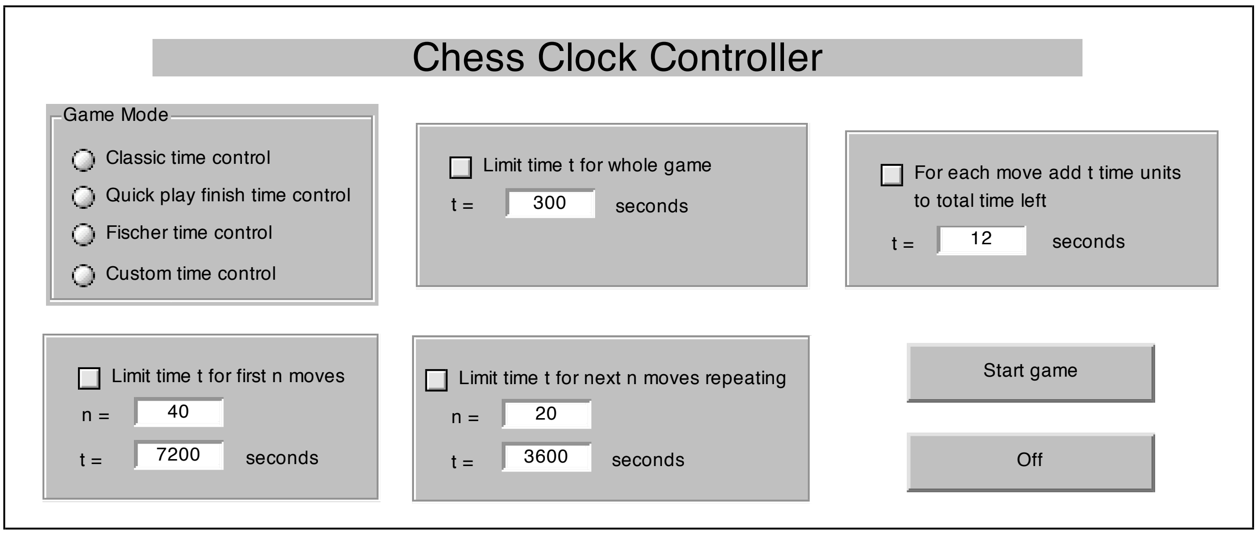

on
white
tick[time_white>0]/
time_white:=time_white−1
1
white_running
white_moves[in(init_white)]/
moves_white:=moves_white−1
white_moves[in(rest_white)]
black_moves
tick[time_white==0]/
black_wins
white_pausing
mode_white
2
init_white
white_moves
[moves_white==0]/
time_white:=time_white
+60*60
rest_white
black
tick[time_black>0]/
time_black:=time_black−1
3
black_running
black_moves[in(init_black)]/
moves_black:=moves_black−1
black_moves[in(rest_black)]
white_moves
tick[time_black==0]/
white_wins
black_pausing
mode_black
4
init_black
black_moves
[moves_black==0]/
time_black:=time_black
+60*60
rest_black
timeout/tick
t1sec
5
key_on/
time_white:=2*60*60;
time_black:=2*60*60;
moves_white:=39;
moves_black:=39
key_off
off

(c)

.



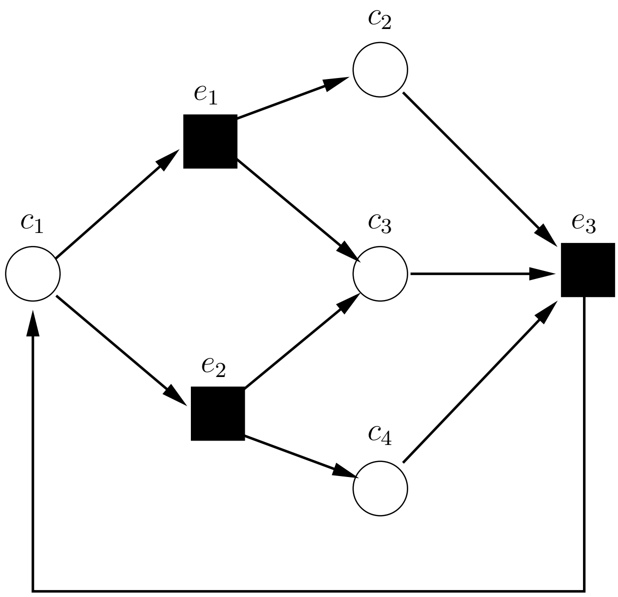
(d)



This is only a sample solution. As this exercise was meant as a creative work, your one may look completely different. You could also have drawn a control automaton which reads different variables from the user and then starts the appropriate game mode, for example.

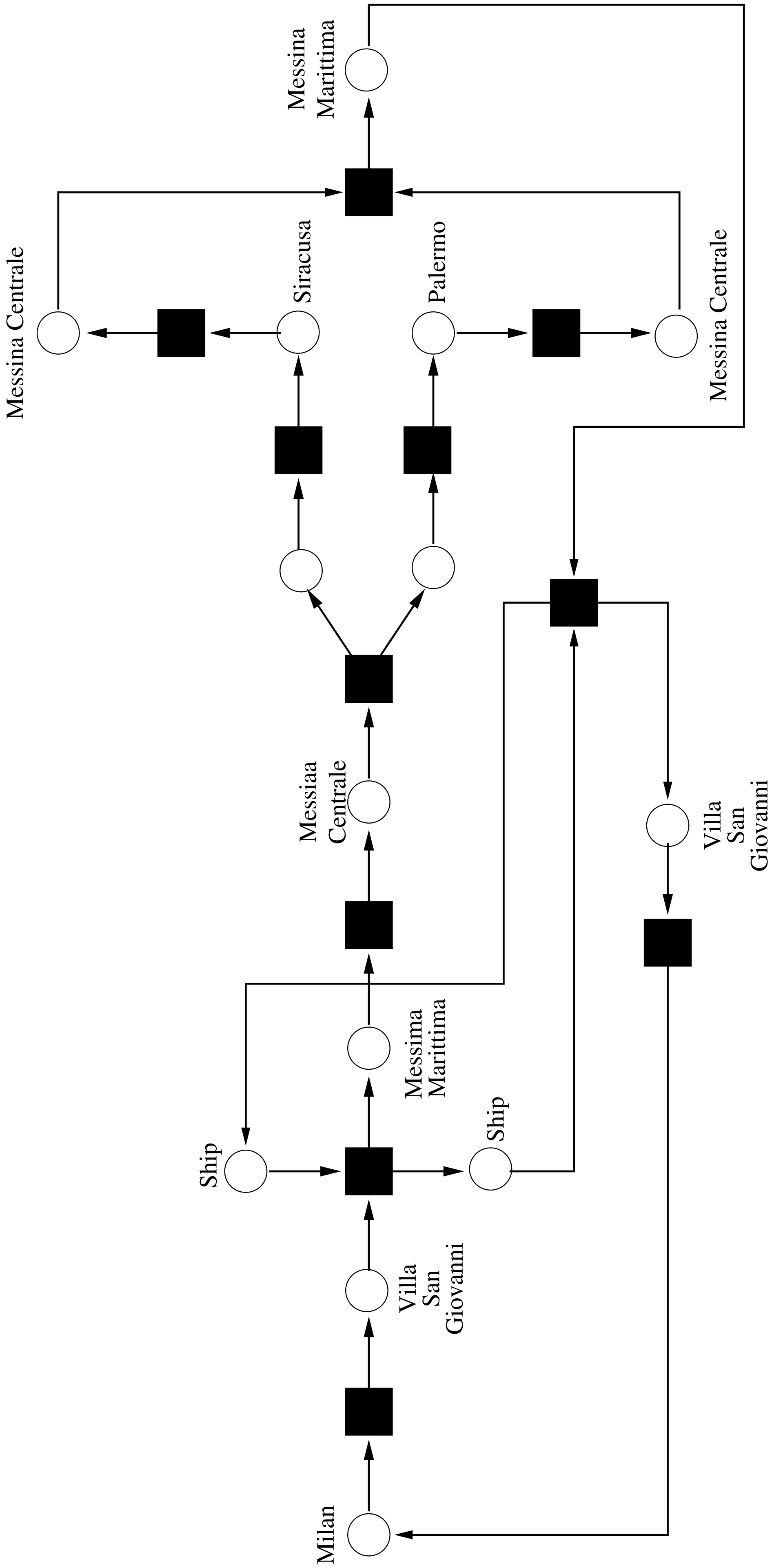
Problem 2 (Petri nets) 10 points

Drawn Petri net:

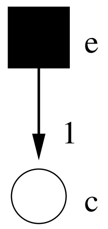


Preconditions of e_3: $\{c_2, c_3, c_4\}$
Postconditions of e_1: $\{c_2, c_3\}$
If you look at two different elements $a, b \in E$, then you will always find that either their pre- or their postcondition sets differ. For example, e_1 and e_2 both have precondition set $\{c_1\}$ but c_2 is not in the postcondition set of e_2 but it is in the one of e_1. So, N is simple.
If you look at some element $(a, b) \in F$, you see that always $(b, a) \notin F$ holds which means that there are no loops, so N is pure.

Problem 3 (Petri nets) **10 points**



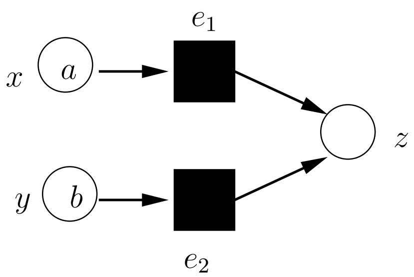
Problem 4 (Petri nets) 10 points



Consider the Petri net given above. It has $M(c) = 0$. However it can reach $M(c) = n$ in n steps for each $n \in \mathbb{N}$ of which there exists infinitely many.

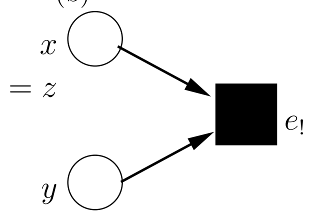
Problem 5 (Petri nets) 10 points

(a)



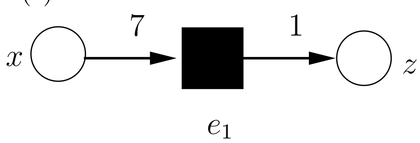
To do the calculation, just fire t_1 and t_2 in an arbitrary order till a deadlock is reached. Then, $M(x) = M(y) = 0$ and $M(z) = a + b$.

(b)



Note that we here have $x = z$, that is the result is at the same place as one of the input parameters. If we fire e_1 as often as possible, we will finally run into a deadlock. If $a \geq b$, then we will have $a - b$ in z, else 0.

(c)



As we run into a deadlock by firing e_1 repeatedly, we will have $\lfloor \frac{x}{7} \rfloor$ tokens on z.